

EMULATION OF INFRASTRUCTURE UNIT FOR INTELLIGENT TRANSPORT SYSTEM

Juraj Gierl

Master Degree Programme (2), FEEC BUT

E-mail: xgiert01@vutbr.cz

Supervised by: Miroslav Balík

E-mail: balik@feec.vutbr.cz

Abstract: This article describes a characteristics of Cooperative Intelligent Transport Systems (C-ITS) and development of an application used to emulate Road Side Unit (RSU) for proof of concept and testing purposes. Application enables users to easily define common messages used in C-ITS such as intersection geometry and it's traffic lights signals and phases.

Keywords: DSRC, ETSI ITS-G5, C-ITS, MAP, RSA, RSU, SPaT, V2X, V2I, V2V

1 ÚVOD

Inteligentné transportné systémy (ITS) popisujú možnosti a metódy pre zlepšenie dopravnej situácie. V súčasnej dobe ale chýbajú prostriedky pre vybudovanie provizorných infraštruktúr, pomocou ktorých by bolo možné súčasti systémov otestovať. Preto vznikla požiadavka pre vytvorenie aplikácie, ktorá túto funkcionálnosť ponúkne.

Cieľom tejto práce je vytvoriť aplikáciu, pomocou ktorej bude môcť užívateľ jednoducho vytvárať geometrie križovatiek a definovať stavy svetelnej signalizácie. Užívateľom vytvorené konfigurácie sú aplikáciou spracované do správ a vyslané do prostredia kooperatívnych systémov inteligentnej dopravy (C-ITS).

2 KOOPERATÍVNY SYSTÉM INTELLIGENTNEJ DOPRAVY

Kooperatívny systém inteligentnej dopravy stavia na komunikácii medzi vozidlami (V2V – Vehicle to Vehicle), vozidla a infraštruktúry (V2I – Vehicle to Infrastructure), vozidla a chodca (V2P – Vehicle to Pedestrian) a mnoho ďalších. Všetky varianty komunikácie sa označujú ako vozidlo ku všetkému (V2X – Vehicle to Everything). Pre prenos dát v systémoch C-ITS je v 5 GHz pásme vyhradených sedem kanálov a ich použitie popisuje štandard IEEE 802.11p [3]. Hlavnou myšlienkou tejto komunikácie je výmena užitočných dát, ktoré môžu pomôcť k prechádzaniu rôznym nehodám. Dáta, ktoré sa vymieňajú sú rôzne, vozidlá zvyčajne vysielajú údaje o svojej polohe, smere a rýchlosti, ktorou sa pohybujú. V prípade, že sa trajektórie dvoch vozidiel prekryjú, vodičom sa včas zobrazí upozornenie na túto situáciu, aby mohol adekvátne zareagovať a predísť kolízii. Existuje niekoľko ďalších upozornení viac v článku [4].

Keďže plný potenciál táto technológia nadobudne, až keď každý účastník cestnej premávky bude vybavený prostriedkami pre komunikáciu so svojim okolím, výrobcovia sa sústreďujú na vývoj infraštruktúrnych jednotiek (RSU – Road Side Unit). Niektoré z veľa rôznych správ, ktoré môže jednotka RSU vyslať, sú napríklad správy o geometrii križovatky a jej svetelných signálov. Na základe týchto správ si vozidlo môže určiť, napríklad či s aktuálnou rýchlosťou prejde bezpečne cez križovatku, alebo už môže spomalovať, pretože v čase príchodu ku križovatke už bude svietiť červená a vozidlo nebude môcť prejsť. Všetky formáty a typy správ sú popísané v štandarde SAE J2735 [1].

3 APLIKÁCIA PRE DEFINOVANIE KRIŽOVATIEK

Aplikácia umožní užívateľovi vytvoriť geometriu akejkoľvek križovatky, ku ktorej je možné pridať informácie o signáloch a fázach svetelnej signalizácie. Aplikácia taktiež poskytuje rozhranie pre definovanie a spúšťanie výstražných správ (RSA – Road Side Alert). Jednotka RSU, na ktorej bude aplikácia spustená je typ vstavaného počítača, ktorý je osadený V2X kartou, GPS modulom a Ethernetovým rozhraním. Aplikácia je typu klient-server, toto riešenie je vybrané na základe požiadavkov na aplikáciu a možností pripojenia k RSU jednotke.

3.1 SERVEROVÁ ČASŤ APLIKÁCIE

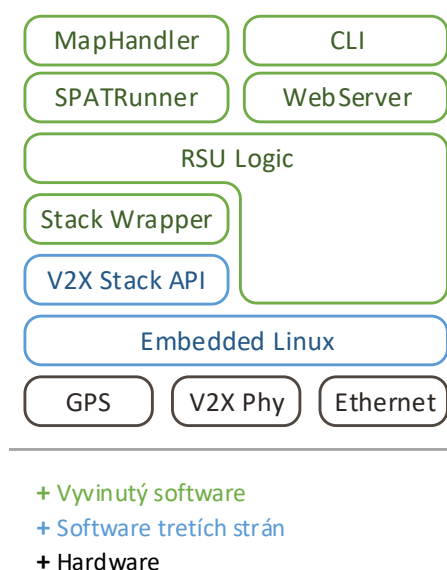
Serverová časť aplikácie má za úlohu vytvoriť a udržiavať aktuálne informácie v správach, ktoré sa vysielajú pomocou V2X rozhrania. Údaje pre vytvorenie týchto správ sú poskytnuté v konfiguračnom súbore, ktorý vygeneruje klientska časť programu z užívateľom zadanych dát. Na obrázku 1 možno vidieť všetky základne bloky RSU jednotky a aplikácie, ktorej bloky sú vyznačené zelenou farbou. Aplikácia je napísaná v programovacom jazyku Python. Jazyk Python bol zvolený pre dobrú prenositeľnosť kódu, rýchli a pohodlný vývoj a ladenie napísaných modulov aplikácie.

Aplikácia musí obsluhovať viacero prostriedkov a spojení naraz, preto sú jednotlivé bloky napísané asynchrónne, čo umožňuje tieto bloky spúšťať súbežne a nezávislo na sebe. Pre tento typ implementácie je v jazyku Python dostupná knižnica *asyncio*, ktorá disponuje prostriedkami pre pseudo-súbežné vykonávanie blokov. Každý asynchrónne napísaný blok sa nazýva korutina (coroutine). Korutiny sa postupne pridávajú do jednej slučky pre beh programu. Táto slučka sa spustí v jednom vlákne, čo spustí beh programu. V jednej slučke je vždy spustená len jedna korutina, ktorá musí úlohu vždy uvoľniť svoje prostriedky, aby sa spustila ďalšia korutina. Jedná sa teda o spôsob jednoduchého kooperatívneho multitaskingu.

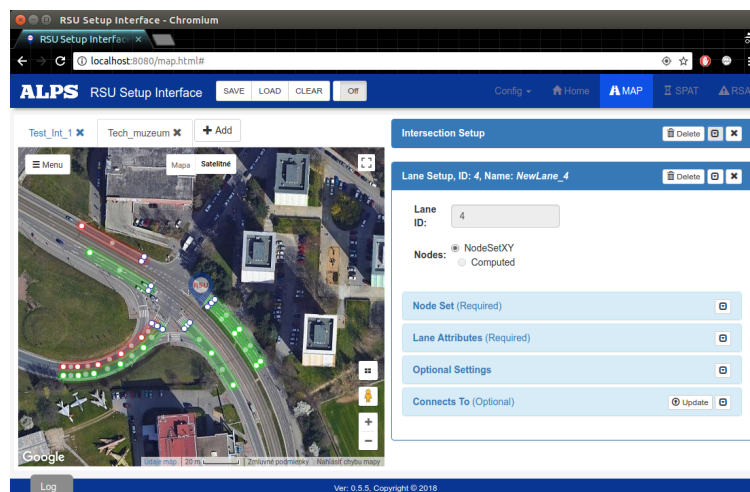
Zelené bloky na obrázku 1 znázorňujú jednotlivé funkčné moduly serverovej časti aplikácie. Modul *RSU* reprezentuje hlavný modul, a disponuje prostriedkami pre manažment V2X správ a spracúva konfiguračný súbor, ktorého ucelené časti predáva ostatným modulom, ktoré sú za vykonanie konkrétnej konfigurácie zodpovedné. Modul *SPATRunner* zodpovedá za vytváranie správ obsahujúce údaje o stavoch a fázach svetelnej signalizácie. Modul *MapHandler* má podobnú úlohu ako predchádzajúci modul, s tým rozdielom, že spracúva geometriu zvolenej križovatky. Moduly *CLI* a *WebServer* sprostredkujú rozhranie pre ovládanie jednotky RSU.

3.2 KLIENSKÁ ČASŤ APLIKÁCIE

Klientská časť programu je realizovaná pomocou webového užívateľského rozhrania a má za úlohu vygenerovať konfiguračný súbor pre serverovú časť. Pomerne rozsiahla štruktúra programu napísaná v JavaSkripte umožňuje užívateľovi pohodlne manipulovať s rôznymi objektami a meniť ich vlastnosti. Príkladom môže byť modul *MAP*, zobrazený na obrázku 2. Pomocou tohoto modulu je užívateľovi umožnené vytvoriť geometriu križovatky a nastaviť všetky povinné a nepovinné para-



Obr. 1: Stavebné bloky RSU jednotky.



Obr. 2: Uživatelské rozhranie pre definovanie geometrie križovatky.

metre, pre každý objekt. Objekt v tomto prípade môže byť napríklad pruh. Klientska časť generuje veľkú časť dát pre konfiguračný súbor automaticky, užívateľ doplní len najnutnejšie dáta. Príkladom môže byť jednoduchá svetelná križovatka typu „X“, ktorú užívateľ vytvorí niekoľko málo klikmi a vo výsledku sa pre systém V2X vygeneruje približne 200 parametrov.

Klientska časť sa môže použiť aj samostatne, bez nutnosti komunikácie so serverom a vytvorené konfiguračné súbory je možné uložiť lokálne a neskôr podľa potreby otvoriť požadované konfigurácie. Viac o funkcionalite klientskej časti možno nájsť v semestrálnej práci [2].

4 ZÁVER

Bola vytvorená aplikácia, ktorá jednoduchým spôsobom umožní užívateľovi vytvoriť a vyslať komplexné správy obsahujúce informácie o geometrii a stavoch svetelnej signalizácie križovatky pre kooperatívny systém inteligentnej dopravy. Aplikácia taktiež umožňuje definovať rôzne výstražné správy. Aplikácia bola sprevádzkovaná na testovacej V2X jednotke firmy ALPS Electric. V súčasnej dobe je používaná ako demonštrátor a prebiehajú diskusie o jej použití v pilotných projektoch v oblasti C-ITS v Českej Republike. Vďaka modulárnemu prístupu je možné aplikáciu jednoducho rozšíriť o ďalšie funkčné bloky.

LITERATÚRA

- [1] Dedicated Short Range Communications (DSRC) Message Set Dictionary. Surface Vehicle Standard J2735, SAE International, 2016
- [2] GIERTL, J. *Emulace infrastrukturní jednotky pro systém inteligentní dopravy*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2018. 43 s. Vedoucí semestrální práce Ing. Miroslav Balík, Ph.D.
- [3] IEEE Std. 802.11p-2010. *IEEE Standard for Information technology—Telecommunications and information exchange between systems—Specific Requirements—Part 11: Wireless AN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amendment 6: Wireless Access in Vehicular Environments*.
- [4] Sjöberg, K.; Andres, P.; Buburuzan, T.; aj.: Cooperative Intelligent Transport Systems in Europe: Current Deployment Status and Outlook. *IEEE Vehicular Technology Magazine*, rok 12. 2. 2017: s. 89–97.